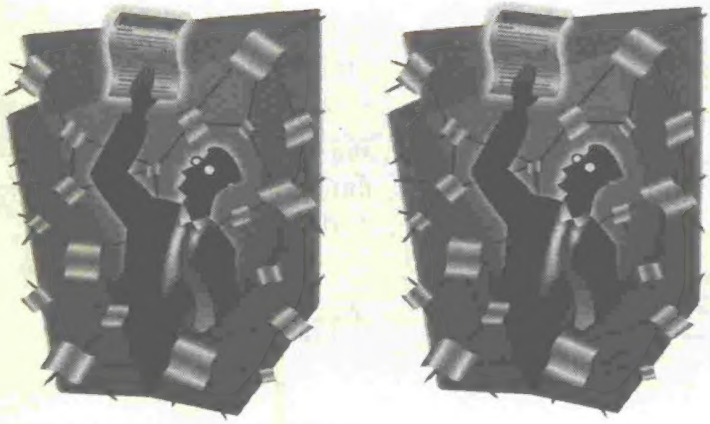
Delphi

บุญเลิศ เอี่ยมทัศนา

Delphi 2.0 กับอินทราเน็ต : เน็ตที่เป็นจริง



สำหรับผู้เขียนโปรแกรมด้วยเดลไฟ โดยไม่จำเป็นต้องเกี่ยวข้องกับอินเทอร์เน็ต และสำหรับการพัฒนาโปรแกรม อาจกระทำได้แม้แต่มีเครื่องเพียงเครื่องเดียว

หากคุณคอยที่จะจัดระบบเครือข่ายอินทราเน็ตภายในองค์กรหรือแม้แต่ในบ้านของคุณ บทความนี้อาจช่วยให้ไม่ต้องคอยต่อไป และเพราะเป็นอินทราเน็ต บทความนี้จึงสำหรับผู้เขียนโปรแกรมด้วยเดลไฟโดยทั่วไป โดยไม่จำเป็นต้องเกี่ยวข้องกับอินเทอร์เน็ต และสำหรับการพัฒนาโปรแกรมอาจกระทำได้แม้แต่มีเครื่องเพียงเครื่องเดียว

กล่าวนำ

ในบทความตอนนี้ ตั้งใจกล่าวถึงการดาวน์โหลดโปรแกรม Web Site มาติดตั้งเป็น Web Server ของเครือข่ายอินทราเน็ต ตามด้วยการโปรแกรม แต่มีบางเรื่องที่จะขอนำมากล่าวก่อนเป็นการทบทวนด้วย

เวบเน็ต

เรื่องแรกคือขอใช้คำว่า "เวบเน็ต" เป็นคำรวมของอินเทอร์เน็ต (Internet) กับอินทราเน็ต (Intranet)

และคำว่า "เวบ" หมายถึงเครือข่ายใยแมงมุม (Web) ไม่จำเป็นต้องเป็นเครือข่ายใยแมงมุมโลก (WWW : World Wid Web) ตามความหมายที่ใช้กันเดิม

ค่า URL

เช่นโฮมเพจของผมมีค่า URL (Uniform Resource Location) เป็น

```
www.inet.co.th/cybermall/newlook
```

ซึ่งจะเรียกสั้น ๆ ว่า MyHomePge โดยที่ค่า URL ที่ไม่ปิดท้ายด้วยชื่อไฟล์เช่นนี้จะหมายถึงไฟล์ index.html เช่นเมื่ออ้างถึงเป็น

```
http://MyHomePge/
```

จะหมายถึง

```
http://MyHomePge/index.html
```

โดยที่ index.html คือไฟล์เอกสาร HTML ซึ่งผมจะต้องเก็บไว้ในไดเรคทอรีที่เป็นโฮมเพจของผม ที่ศูนย์บริการอินเทอร์เน็ต และเรียกไฟล์นี้ว่าโฮมเพจด้วย (อย่างไม่ถูกต้องนัก)

เพื่อใช้เชื่อมโยงไปยังเวบเพจอื่น ๆ โดยเฉพาะกับเวบเพจอื่นของผมเอง โดยอ้างถึงค่า URL ของเวบเพจนั้น (ตรงที่ทำเป็นข้อความหรือภาพให้เลือก)

ดังเช่นถ้าผมเก็บเรื่องของสุนัขไว้ในไฟล์ dog.htm จะได้ค่า URL ของเวบเพจนี้ว่า

```
MyHomePge/dog.htm
```

และถ้าผมสร้างไดเรคทอรีรองที่โฮมเพจ เช่นให้เป็น animal ก็จะได้ค่า URL เป็น

```
MyHomePge/animal/
```

ซึ่งหมายถึงไฟล์ index.html ในไดเรคทอรีใหม่นี้เช่นกัน และถ้าผมย้ายไฟล์ dog.htm มาไว้ในไดเรคทอรีนี้ จะได้ค่า URL เป็น

```
MyHomePge/animal/dog.htm
```

จะเห็นว่าในค่า URL จะต้องมีชื่อไฟล์ (รวมทั้งไดเรคทอรี) และไม่เพียงแค่ไฟล์ .htm เท่านั้น ยังมีไฟล์อื่นอีก

คิดง่าย ๆ (ไม่จำเป็นต้องถูกต้องนัก) อย่างเช่นคนที่อเมริกาต้องการเยี่ยมชมโฮมเพจของผม อ้างถึงค่า URL มา ก็จะเท่ากับเรียกหรือเชื่อมการติดต่อมายัง www.inet.co.th ซึ่งก็คือค่า URL ของศูนย์บริการ INET เพื่อขอไฟล์ index.html ในไดเรคทอรี /cybermall/newlook (ซึ่งความจริงไม่ใช่ไดเรคทอรีนี้โดยตรง) แต่มองค่า URL ทั้งหมดเป็นของโฮมเพจของผม

ค่า IP Address

บางเวบเพจจะให้ค่า URL เป็นค่า IP Address เช่น

```
http://www.msn.com/
```

กับ

```
http://207.68.142.58/
```

จะให้ผลเช่นเดียวกัน ไม่มีหนังสือเล่มใดเขียนไว้ว่า "ค่า URL คือค่า IP Address ตามด้วยชื่อไดเรคทอรี (หากมี) และชื่อไฟล์" เข้าใจว่าคงไม่ถูกต้อง แต่เป็นเช่นนี้ตามที่จะนำไปใช้ต่อไปด้วยครับ

ค่า URL เทียม

เราสามารถสร้างเวบเพจส่วนตัวเก็บไว้ที่เครื่องของเราได้ เช่น ทำเป็นบันทึกการเดินทางเก็บค่า URL ของเวบเพจที่สนใจไว้ในไฟล์

```
c:\mypage\log.htm
```

สามารถจะใช้บราวเซอร์เช่น Netscape (Netscape Navigator) อ่านคือบราวซ์ (browse) ดูได้ เช่นโดยเลือกเมนู

```
File/Open File
```

หรือป้อนค่าซึ่งขอเรียกเป็นค่า URL เทียม ตรงกรอบ Location ซึ่งจะต้องป้อนเป็น

```
file:///CI/mypage/log.htm
```

ที่นำมากล่าวเพราะจะต้องอ้างถึงต่อไปครับ และจะขออ้างถึง Netscape เป็นตัวแทนของบราวเซอร์

โปรแกรมกับเวบเน็ต

สามารถแยกการเขียนโปรแกรมกับเวบเน็ตได้เป็น 2 ภาค 3 เรื่อง คือ

1. ภาคไคล์เอ็นต์ เช่น เขียนโปรแกรมทำหน้าที่เป็นบราวเซอร์ดังที่เคยนำมากล่าวบ้างแล้ว

2. ภาคเซอร์ฟเวอร์ แยกเป็น

- เขียนโปรแกรมเป็น Web Server ดังที่จะดาวน์โหลด WebSite มาใช้

- เขียนโปรแกรมสร้างเวบเพจ

การสร้างเวบเพจ

ก็คือการสร้างเอกสาร HTML ซึ่งก็คือไฟล์ .html หรือ .htm ซึ่งแบ่งออกเป็น 2 กรณีคือ

1. ถ้าเป็นเอกสาร HTML ธรรมดา มีแค่การเชื่อมโยงไปยังเวบเพจอื่น เราจะสร้างเวบเพจนี้ไว้ ณ ที่ใดคือจะเก็บไฟล์นี้ไว้ ณ ที่ใดก็ได้ เช่นที่โฮมเพจที่ศูนย์บริการอินเทอร์เน็ต

2. ถ้าเป็นเอกสาร HTML ที่เชื่อมโยงไปยัง CGI Script จะต้องเก็บไว้ที่เวบไซท์ที่เป็นเวบเซอร์ฟเวอร์เท่านั้น ซึ่งก็คือที่เครื่องที่สำนักงาน หรือที่บ้านที่ติดตั้งโปรแกรมเวบเซอร์ฟเวอร์ โดยหลักใหญ่ก็คือเพื่อการให้บริการในรูปแบบต่าง ๆ

อินเทอร์เน็ตหรืออินทราเน็ต

จึงสุดแต่จะให้บริการในเครือข่ายใด หากต้องการให้บริการทางอินเทอร์เน็ต ก็จะต้องต่อเชื่อมเครื่องที่เป็นเซอร์ฟเวอร์เข้ากับอินเทอร์เน็ต เช่นผ่านทางศูนย์บริการอินเทอร์เน็ต ซึ่งต้องเสียค่าใช้จ่ายสูงพอสมควร (ตั้งแต่หมื่นบาทถึงล้านบาทต่อเดือน ขึ้นอยู่กับขนาดของบริการ)

แต่ถ้าเป็นอินทราเน็ตคือเครือข่ายในองค์กร ซึ่งถ้ามีเครื่องที่ติดต่อถึงกันเป็นเครือข่ายด้วยวินโดวส์ 95 หรือ NT หรือผสมกัน และได้ติดตั้ง TCP/IP (ตามที่จะกล่าวต่อไป) ไว้แล้ว ก็สามารถกำหนดคือติดตั้งโปรแกรมให้เครื่องหนึ่งเครื่องใดหรือหลายเครื่องเป็นเวบเซอร์ฟเวอร์ได้ทันที

จึงขอสมมติว่าขณะนี้มีเวบเซอร์ฟเวอร์แล้ว ไม่ว่าจะเป็นอินเทอร์เน็ตหรืออินทราเน็ต และเมื่อมีเวบเซอร์ฟเวอร์ก็จะเก็บเอกสาร HTML ทุกชนิดไว้ที่เวบเซอร์ฟเวอร์

CGI Script

ผมขอเลยไปถึง CGI Script ก่อน กล่าวคือ CGI คือ Common Gateway Interface เพื่อให้สามารถเชื่อมโยงเอกสาร HTML กับโปรแกรมในเครื่องของเรา (ที่เป็นเวบเซอร์ฟเวอร์) ได้ทาง CGI Script ซึ่งก็คือโปรแกรม เช่นเมื่อเขียนสคริปต์ด้วยเคลไฟก็คือไฟล์ .exe หนึ่ง ๆ ซึ่งจะบ่งชื่อไว้ในเอกสาร HTML

ความสัมพันธ์

และขอต่อไปถึงความสัมพันธ์ครับ (ซึ่งจะขอใช้คำเป็นภาษาอังกฤษเพื่อให้สะดุดตา) ระหว่าง Browser-Server-CGI Script กล่าวคือ

เมื่อผู้ใช้ป้อนค่า URL ตรงกรอบ Location แล้วกดคีย์ Enter หรือเลือกข้อความหรือภาพที่จัดไว้ให้เลือก Browser จะนำค่า URL มาอ้างถึงคือเชื่อมโยงไปยัง Web Server ตามค่า URL นั้น เพื่อส่งคำขอไฟล์ไป

ซึ่งถ้าเป็นไฟล์ HTML เช่น dog.htm ทาง Web Server ก็จะส่งไฟล์นี้ย้อนกลับไปให้แก่ Browser

แต่ในบางเวบเพจจะเห็นว่ามีแบบฟอร์มให้กรอก ตรงนี้ครับที่จะทำให้เกิดการเชื่อมโยงไปยัง CGI Script คือเมื่อผู้ใช้เลือกปุ่มเช่น Submit จะเกิดการเชื่อมโยงดังนี้

1. Browser จะส่งคำขอไฟล์เช่นกัน แต่เป็นไฟล์ .exe (หรืออาจนามสกุลอื่นอีก) ซึ่งจะขออ้างชื่อเป็น abc.exe พร้อมทั้งส่งข้อมูลที่ผู้ใช้กรอกให้ไปด้วย

2. Web Server จะดำเนินการดังนี้

2.1 นำข้อมูลที่ได้รับมา เขียนลงไฟล์ .ini สมมติให้ชื่อเป็น xyz.ini ตามแบบฉบับของไฟล์ .ini เช่น

```
[Form Literal]
Name=Boonlert
```

ซึ่งต่อไปจะเห็นว่าคำว่า Name ทำหน้าที่คล้ายตัวแปร

2.2 รันโปรแกรม abc.exe พร้อมทั้งผ่านชื่อไฟล์ xyz.ini เป็นค่าคอมมานด์ไลน์

3. CGI Script ก็คือเราผู้เขียนโปรแกรมเป็น CGI Script จะต้องดำเนินการดังนี้

3.1 อ่านข้อมูล จากไฟล์ xyz.ini นำมาดำเนินกรรมวิธี ซึ่งเป็นวิธีเขียนโปรแกรมตามปกติ

3.2 หากต้องส่งผลกลับจะต้อง

- อ่านชื่อไฟล์ Output File ซึ่ง Web Server จะให้มาด้วย ใน xyz.ini สมมติว่าให้ชื่อเป็น outfi.htm
- นำไฟล์ outfi.htm มาสร้างเป็นไฟล์ HTML ซึ่งจะทำให้เกิดเป็นเวบเพจใหม่อีก 1 เพจ
- ยุติการรัน

4. Web Server จะนำไฟล์ outfi.htm ส่งกลับ

5. Netscape นำออกแสดงเช่นเดียวกับไฟล์ HTML ทั้งหลาย

ด้วยวิธีง่าย ๆ เช่นนี้ทำให้เอกสาร HTML ซึ่งปกติไม่ผูกพันกับสิ่งใด สามารถติดต่อกับเครื่องคือติดต่อกับโปรแกรมได้ และส่วนใหญ่ก็คือเพื่อติดต่อกับฐานข้อมูล

แต่ด้วยความที่ผมพึ่งเตาะแตะเข้าสู่วงการนี้จับต้นชนปลายไม่ถูก กว่าจะได้ข้อความ 5 ข้อนี้ ร่วมเดือนครับ

สิ่งที่ต้องมี

ผมพอใจที่นำเรื่องนี้มากล่าว เพราะในขั้นต้นนี้ไม่ต้องการงบประมาณใดเลย ถ้าคุณมีสิ่งต่าง ๆ ต่อไปนี้อยู่แล้วคือ

1. เครื่องจำนวนหนึ่ง ติดตั้ง Windows 95 หรือ NT หรือผสมกัน โดยติดต่อถึงกันเป็นเครือข่ายท้องถิ่นอยู่แล้ว (ก็คือตามสภาพที่มีอยู่) โดยที่อาจกำหนดให้เครื่องหนึ่งหรือหลายเครื่องเป็น Web Server ซึ่งต่อไปจะขอเรียกว่าเครื่องแม่ ส่วนเครื่องอื่นจะขอเรียกเป็นเครื่องลูก เพื่อหลีกจากคำว่า client/server ซึ่งให้ความหมายหนักไปทางระบบงานฐานข้อมูล

แต่เพื่อการศึกษา ทดลองใช้งาน หรือแม้แต่เพื่อเตรียมโปรแกรมที่บ้านเพื่อนำไปใช้ที่สำนักงาน สามารถใช้เครื่องเพียงเครื่องเดียวได้

2. โปรแกรม Web Server ซึ่งผมจะใช้ WebSite ด้วยเหตุผลหลายประการครับ เช่น ใช้ง่ายราคาถูก ใช้วินโดวส์ 95 เป็น Web Server ได้ และข้อสำคัญอีกเรื่องหนึ่งคือให้เราสามารถทดลองใช้งานได้โดยยังไม่มีข้อจำกัดใด ตามที่จะดาวน์โหลดขอยืมเขามาใช้ก่อน

3. Delphi 2.0 ก็เพราะผมจะใช้เขียนโปรแกรม และบทความนี้สำหรับผู้ที่เขียนโปรแกรมด้วยเดลไฟ ซึ่งถือว่าต้องมีอยู่แล้ว

และจะต้องรู้วิธี (เขียนโปรแกรม) สร้างเอกสาร HTML เช่นตามที่ผมนำมากล่าวในนิตยสาร Internet เล่ม 3 ในเรื่อง "เตาะแตะถึง HTML" ก็น่าจะพอ หรือดูตัวอย่างได้ที่โฮมเพจของผมโดยเลือกหัวข้อ "Write HTML Script"

ดาวน์โหลด

จุดนี้จุดเดียวครับที่ผู้ที่ไม่ได้ติดต่อกับอินเทอร์เน็ต คงต้องอาศัยเขาดาวน์โหลดซึ่งเป็นปัญหาหนึ่ง อีกปัญหาหนึ่งคือเมื่อดาวน์โหลดได้มาแล้วจะนำมาติดตั้งที่เครื่องของคุณได้อย่างไร เพราะโปรแกรมมีขนาดใหญ่ คงต้องใช้โปรแกรมติดต่อไฟล์ เช่น arj.exe

ในการดาวน์โหลดความจริงมีโปรแกรม WebSite ที่จะดาวน์โหลดเพียงโปรแกรมเดียว แต่จะขอแนะนำให้ดาวน์โหลดโปรแกรมอื่นด้วยเพื่อช่วยในการทำงาน จะดาวน์โหลดโปรแกรมต่าง ๆ ได้ดังนี้

1. WebSite 1.1 ของ O'Reilly ได้ที่

```
website.ora.com
```

และขอให้ดาวน์โหลดคู่มือในการติดตั้งด้วยที่

```
software.ora.com/techsupport/doc.html
```

2. HomeSite เพื่อใช้ในการสร้างเอกสาร (เขียนโปรแกรม) HTML ที่

```
www.dexnet.com/homesite.html
```

3. IntraBuilder ของบอร์แลนด์เผื่อต้องการทดลองสร้างเอกสาร HTML ด้วย JavaScript ที่

```
http://netserv.borland.com/intrabuilder/download.html
```

4. HTML Reference Library ของ Stephen Le Hunte ขอให้ดูจากคำอธิบายในโฮมเพจของผมครับ เพราะมีหลายที่ด้วยกัน

และทั้งหมดนี้จะนำไปไว้ที่โฮมเพจของผม ขอให้เริ่มจากหัวข้อ "Build your own Intranet"

การจัดระบบอินทราเน็ต

เมื่อได้โปรแกรมมาแล้วก็สามารถจะจัดระบบอินทราเน็ตได้ทันทีครับ ซึ่งหมายถึงว่าคุณมีเครื่องที่ติดตั้ง Windows 95 หรือ NT เป็นเครือข่ายอยู่แล้ว หรืออาจมีเพียงเครื่องเดียว จะจัดระบบได้โดยติดตั้งโปรแกรมเพียง 2 ตัวเท่านั้นคือ

1. TCP/IP
2. WinSite

ซึ่งจะต้องติดตั้ง TCP/IP ก่อน และต่อไปอาจเสริมด้วยโปรแกรมอื่นได้อีกตามต้องการ

ติดตั้ง TCP/IP

ผมได้กล่าวถึงการติดตั้ง TCP/IP ไว้ในไมโครฯ 133 หากไม่มีก็คงต้องดูที่โฮมเพจของผมครับ ซึ่งจะต้องติดตั้งให้แก่เครื่องในเครือข่ายทุกเครื่อง

ขอขยายความว่าค่า IP Address เนื่องจากไม่ได้ติดต่อกับอินเทอร์เน็ต จึงสุดแต่จะตั้ง ซึ่งควรมีผู้กำหนด

และขอให้เตรียมชื่อเพื่อใช้แทน IP Address แต่ละตัวด้วย เช่น person.abc คือฝ่ายบุคคล บริษัท abc เนื่องจากต้องป้อนแต่ผมยังไม่รู้จักวิธีใช้ จึงควรเตรียมให้เป็นระบบไว้ก่อน และสำหรับใช้เป็นอีเมล์แอดเดรสด้วยเช่นใช้เป็น boonlert@person.abc

การติดตั้ง WebSite

ทำตามคู่มือที่ดาวน์โหลดมา โดยรันโปรแกรม website11.exe และเพราะผมติดตั้งกับ Windows 95 ซึ่งติดตั้งได้ง่ายจึงคงทำตามคำแนะนำของโปรแกรมติดตั้ง ไม่เปลี่ยนไม่แก้สิ่งใด ป้อนแต่ค่าที่เตรียมไว้

ส่วนการที่จะติดตั้ง WebSite ให้แก่เครื่องใดเพื่อใช้เป็น Web Server สุดแต่พิจารณาครับ หากมี NT ในเครือข่ายก็ควรติดตั้งที่ NT และอาจติดตั้งให้แก่หลายเครื่องได้ เช่นที่ ฝ่ายบุคคลเครื่องหนึ่ง ฝ่ายขายเครื่องหนึ่ง เป็นต้น

เมื่อติดตั้งแล้วขอให้ทดสอบด้วยโปรแกรม Server Self-Text ครับ ของผมไม่มีปัญหาใด จึงไม่มีประสพการณ์

เกือบลืมบอกครับ ก่อนที่จะรันโปรแกรมทดสอบจะต้องมีคือได้ติดตั้งบราวเซอร์ไว้ก่อนแล้ว ของผมใช้ Netscape Navigator บ้าง Microsoft Internet Explorer บ้าง (อยู่ต่างเครื่องกัน) และของผมได้ติดตั้งไว้ก่อนที่จะติดตั้ง WebSite

กำหนดเอเลียส

คุณอาจเคยพบคำว่า cgi-bin มาบ้าง คำนี้เป็นคำที่นิยมใช้กันใน Web Server เป็นเอเลียส (alias) เพื่อใช้แทนชื่อไดเรคทอรี (ที่อาจยาว) ซึ่งเก็บไฟล์ที่เป็น CGI Script ซึ่งส่วนมากได้แก่ไฟล์ .exe

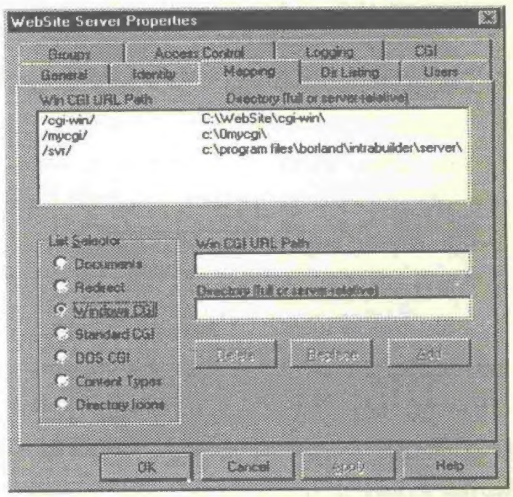
และสำหรับของ WinSite มีอีก 2 คำคือ cgi-win ซึ่งหมายถึง Win CGI Script ที่เป็นเรื่องของวินโดวส์ตามที่กล่าวในหัวข้อ "ความสัมพันธ์"

อีกคำหนึ่งซึ่งจะเกี่ยวกับการเขียนโปรแกรมด้วยเดลไฟคือ wsdocs เป็นเอเลียสของไดเรคทอรีที่ใช้เก็บเอกสาร HTML และไฟล์ที่เกี่ยวข้องต่าง ๆ เช่นไฟล์รูปภาพ

จึงมีปัญหาว่า เราจะเก็บไฟล์ทั้งสองชนิดนี้ไว้ในไดเรคทอรีของ WebSite หรือเก็บไว้ในไดเรคทอรีของเราเอง ผมเลือกวิธีหลังนี้ ซึ่งจะดำเนินการได้โดย

1. กำหนดชื่อที่จะใช้เป็นเอเลียส เช่น ของผมกำหนดเป็น mycgi และ mypage เปิดไดเรคทอรีใหม่ 2 ไดเรคทอรี สำหรับ mycgi เพื่อเก็บไฟล์ .exe ที่เป็น CGI Script และสำหรับ mypage เพื่อเก็บเอกสาร HTML (ซึ่งอาจเปิดไดเรคทอรีรองต่อไปได้ภายหลัง)

2. รัน WebSite โดยเลือกจาก Start/WebSite 1.1/WebSite Server จะแสดงไอคอนเป็นรูปฟันเฟืองอยู่ทางด้านขวาของ Task bar



รูปที่ 1 ไดอะล็อกบ็อกซ์ WebSite Server Properties หน้า Mapping เพื่อการกำหนดเอเลียส

3. คลิ๊กเม้าส์ขวาที่ไอคอนนี้ แล้วเลือกรายการ WebSite server properties จะแสดงไดอะล็อกบ็อกซ์ ให้เลือกหน้า (tab) Mapping แล้วเชคปุ่ม Windows CGI ในกรอบ List Selector จะได้ผลดังรูปที่ 1 เพื่อที่จะกำหนดเอเลียสเป็น mycgi โดย

- ที่กรอบ Win CGI URL Path (กรอบบน) ให้ป้อนข้อความว่า

```
/mycgi/
```

- ที่กรอบ Directory.. (กรอบล่าง) ให้ป้อนไดเรคทอรีที่เตรียมไว้ เช่นของผมป้อนเป็น

```
C:\Omycgi\
```

กำหนดเอเลียสเป็น mypage (ตามของผม) โดย เชคปุ่ม Documents ที่กรอบ List Selector แล้วป้อนเอเลียสกับไดเรคทอรีตามที่กล่าวมา

เสร็จแล้วให้เลือกปุ่ม OK

ตัวอย่างแรก

จะแสดงแบบฟอร์มตามรูปที่ 2 ให้กรอกชื่อและอายุ เมื่อเลือกปุ่ม OK จะแสดงคำตอบดังรูปที่ 3 จะเขียนโปรแกรมได้ดังนี้

เอกสาร HTML

ตามลิสติงที่ 1 แสดงการใช้ แทร็ค FORM ทำให้ได้แบบฟอร์มตามรูปที่ 2 กล่าวคือ

1. ในตัวแท็ก FORM มีค่ารายละเอียดที่ให้กำหนดคือ

- METHOD เพื่อเลือกวิธีการผ่านค่าเป็น GET (ซึ่งควรเลิกใช้) หรือ POST

- ACTION เพื่อระบุชื่อสคริปต์ไฟล์ จะเห็นว่าได้นำเอเลียสมาใช้

2. แท็ก INPUT เทียบเท่ากับกำหนดให้เป็นคอนโทรล มีค่ารายละเอียดเป็น

- TYPE ให้เป็น TEXT เทียบเท่ากับกรอบอีดิต

- NAME เป็นการตั้งชื่อคอนโทรล ซึ่งจะใช้อ้างถึงใน CGI Script

- SIZE เพื่อบ่งความกว้างของกรอบเป็นอักษร

3. ในแท็ก INPUT สุดท้าย

- TYPE ให้เป็น submit ทำให้เป็นปุ่มควบคุม เทียบได้กับปุ่ม OK ซึ่งเมื่อผู้ใช้เลือก Browser จะส่งข้อมูลให้แก่ Server ตามที่กล่าวมา

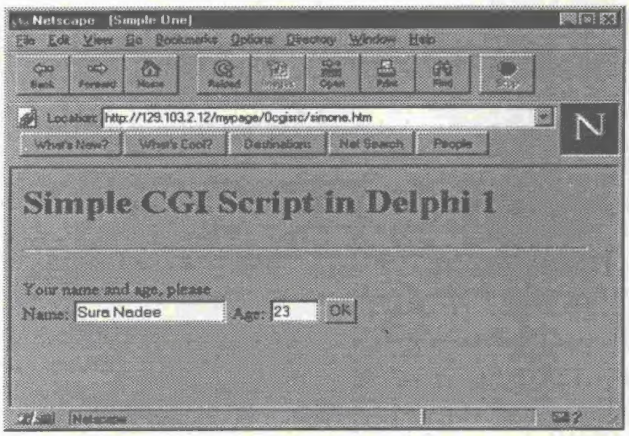
- VALUE คือค่าที่ให้ เท่ากับให้ตั้งชื่อปุ่มนี้ว่า OK

ขอให้ป้อนโปรแกรมตามลิสติงที่ 1 นี้เช่นด้วย NotePad เก็บไว้ใน mypage ซึ่งของผมเก็บไว้ในไดเรคทอรีรองอีกทอดหนึ่งคือใน

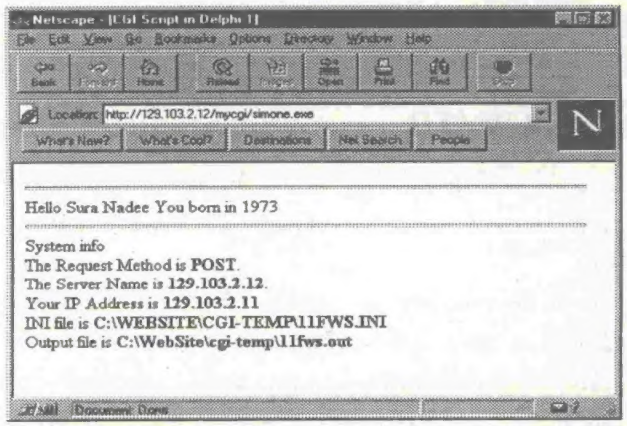
```
/mypage/Ocgisec/
```

CGI Script

ขอให้เปิดใช้เดลไฟ เปิดโปรเจ็กต์ใหม่ ป้อนโปรแกรมตอบสนองอีเวนต์ OnCreate คือ FormCreate ตามลิสติงที่ 2 ซึ่งมีเรื่องที่จะต้องดำเนินการดังนี้



รูปที่ 2 แบบฟอร์มเอกสาร HTML ตามลิสติงที่ 1



รูปที่ 3 แบบฟอร์มเอกสาร HTML ตามที่สร้างด้วย CGI Script ในลิสติงที่ 2

ลิสติงที่ 1

```
<HTML>
  <HEAD> <TITLE>Simple One </TITLE> </HEAD>
  <BODY>
<H1> Simple CGI Script in Delphi 1 </H1> <HR>

<!- ========== Main body ========== ->

  <FORM  METHOD="POST" ACTION="/mycgi/simone.exe
  Your name and age, please <BR>
  Name: <INPUT TYPE=TEXT NAME=Name>
  Age: <input type=TEXT name=Age size=5>
  <INPUT TYPE="submit" VALUE="OK"> <BR>
  </FORM>

 </BODY>
</HTML>
```

1. ให้กำเนิดอินพุตไฟล์ ตามที่ให้กำเนิด oIniFi เป็นออบเจ็กต์ของสกุล TImiFile (เป็นสกุลสำหรับอ่านข้อมูลในไฟล์ .INI โดยทั่วไป) โดยผ่านชื่อไฟล์ .INI (คือ 11FWS.INI ตามรูปที่ 3) ที่ Server ผ่านมาให้ทางคอมมานด์ไลน์ (ParamStr) ไปให้

ซึ่งการดำเนินการในช่วงตอนนี้จะมีการส่งเมสเสจ จึงต้องใช้ ProcessMessage ถอนเมสเสจจากคิว เพื่อให้เกิดผลก่อนที่จะปฏิบัติตามคำสั่งต่อไป

2. อ่านชื่อเอาต์พุตไฟล์ คือที่ให้อ่านมาเก็บไว้ใน OutFiNme (ตามรูปที่ 3 คือ 11fws.out) เพื่อที่จะใช้สร้างเอกสาร HTML ส่งกลับ

นอกจากนี้ได้ให้อ่านค่าต่าง ๆ มาแสดงในรูปที่ 3 เพื่อการศึกษา

3. รับข้อมูลจาก Browser ตามที่ผู้ใช้ส่งมา คือที่อ่านค่าให้แก่ nmeS และ ageS จะเห็นว่าได้อ้างถึงชื่อคอนโทรลคือ Name กับ Age ตามที่ให้ค่าแก่ NAME ในลิสติงที่ 1

เสร็จแล้วให้ยุบ oIniFi เพราะหมดธุระที่จะใช้แล้ว

4. ดำเนินกรรมวิธีข้อมูล ซึ่งสุดแต่ว่าจะเขียนโปรแกรมในเรื่องใด เป็นการเขียนโปรแกรมด้วยเคลไฟตามปกติ ตามตัวอย่างนี้ได้ให้หาค่าปีปัจจุบัน เพื่อนำอายุมาลบเป็นปีเกิด เป็นข้อมูลที่ต้องการส่งกลับไปแจ้งให้ผู้ใช้ทราบ

5. สร้างเอกสาร HTML เพื่อการส่งกลับโดยสร้างในไฟล์ตามชื่อไฟล์ใน OutFiNme จะเห็นว่า

- ตอนต้นเป็นข้อความที่กำหนดให้มี และที่ต้องมีคือ ที่บ่ง content-type เป็น text/html กับที่ให้บรรทัดว่าง 1 บรรทัดคือ

```
Writeln(Outfi, '');
```

ก่อนที่จะถึงตัวเอกสาร HTML

- ต่อไปเป็นการสร้างเอกสาร HTML ตามปกติ เช่นเดียวกับที่ป้อนเช่นให้แก่ NotePad เพียงแต่กระทำด้วยวิธีเขียนลงไฟล์

6. ยุติการรัน เมื่อสร้างเอกสาร HTML เสร็จ หรือในกรณีที่ไม่สร้างคือเมื่อดำเนินกรรมวิธีข้อมูลเสร็จ จะต้องยุติการรันโปรแกรมนี้ทันที ดังที่ผมให้ยุติด้วย Terminate เพื่อส่งคืนการทำงานให้แก่ WebSite เพื่อส่งเอกสาร HTML ที่สร้างใหม่นี้ ให้แก่ Browser และจะเป็นการยุติการติดต่อระหว่าง Browser กับ Server คือ WebSite ด้วย

7. เก็บไว้ใน mycgi ผมตั้งชื่อโปรแกรมนี้ว่า SimOne ให้ชื่อไฟล์เป็น simone1.pas และ simone.dpr ซึ่งเมื่อคอมพายล์แล้ว (รันไม่ได้) ให้ย้ายไฟล์ simone.exe ไปไว้ในไดเรคทอรี mycgi

คำอธิบายเสริม

1. ตามรูปที่ 2 จะเห็นว่าได้อ้างถึงไฟล์ simone.htm ด้วยค่า URL จริง โดยใช้ IP address

2. ตามรูปที่ 3 จะเห็นว่าได้อ่านเอกสาร HTML จากอีกเครื่องหนึ่ง ซึ่งมีค่า IP address เป็น 129.103.2.11

ลิสติงที่ 2

```
unit simone1;

procedure TForm1.FormCreate(Sender: TObject);
var  oIniFi : TIniFile;  IniFiNme : String;
     OutFi : TextFile;  OutFiNme,cmmLne : String;
     ReqMthd,SrvrNme,RmteAddr : String;
     nmeS,ageS,ansS : String;  age,yy,mm,dd : Word;
     today : TDateTime;
begin
  oIniFi := TIniFile.Create(ParamStr(1));
  Application.ProcessMessages;

  OutFiNme := oIniFi.ReadString('System', 'Output File', '');
  ReqMthd := oIniFi.ReadString('CGI', 'Request Method', '');
  SrvrNme := oIniFi.ReadString('CGI', 'Server Name', '');
  RmteAddr := oIniFi.ReadString('CGI', 'Remote Address', '');

  nmeS := oIniFi.ReadString('Form Literal', 'Name', '');
  ageS := oIniFi.ReadString('Form Literal', 'age', '');
  oIniFi.Free;
  age := StrToInt(Trim(ageS));  today := Now;
  DecodeDate(today, yy,mm,dd);
  ansS := ' You born in ' + IntToStr(yy - age);

  AssignFile(OutFi, OutFiNme);  Rewrite(OutFi);
  Writeln(OutFi, 'HTTP/1.0 200 OK');
  Writeln(OutFi, 'Server: WebSite 1.0');
  Writeln(OutFi, 'Content-type: text/html');
  Writeln(OutFi, '');

  Writeln(OutFi, '<HTML><HEAD>');
  Writeln(OutFi, '<TITLE>CGI Script in Delphi 1</TITLE></HEAD>');
  Writeln(OutFi, '<BODY BGCOLOR="#FFFFFF"><HR>');

  Writeln(OutFi, 'Hello ' + nmeS + ansS + '<HR>');

  Writeln(OutFi, 'System info' + '<BR>');
  Writeln(OutFi, 'The Request Method is <B>' + ReqMthd + '</B><BR>');
  Writeln(OutFi, 'The Server Name is <B>' + SrvrNme + '</B><BR>');
  Writeln(OutFi, 'Your IP Address is <B>' + RmteAddr + '</B><BR>');
  Writeln(OutFi, 'INI file is <B>' + ParamStr(1) + '</B><BR>');
  Writeln(OutFi, 'Output file is <B>' + OutFiNme + '</B><BR>');

  Writeln(OutFi, '</BODY></HTML>');

  CloseFile(OutFi);

  Application.Terminate;
end;

end.
```

ลิสติงที่ 3

```
program simone;

uses
  Forms,
  simone1 in 'simone1.pas' {Form1};

{$R *.RES}

begin
  Application.Initialize;
  Application.CreateForm(TForm1, Form1);
  Application.Run;
end.
```

สร้างสกุลของหน้าต่างไว้ใช้

ตามลิสติงที่ 2 จะเห็นว่ามีข้อความหลายตอน ที่จะต้องป้อนเหมือนกันในทุกโปรแกรมและมีเรื่องอื่นอีก ควรทำเป็นสกุลของ Form คือของหน้าต่างเก็บไว้ใช้ ซึ่งจะดำเนินการได้ดังนี้

1. เปิดโปรเจ็กต์ใหม่ แล้วเปลี่ยนชื่อ Form1 เป็น CgiForm โดยให้ค่าแก่ Name

2. ป้อนโปรแกรมตามลิสติงที่ 4 (สำหรับคำอธิบายโปรแกรมจะกล่าวภายหลัง)

3. เตรียมไดเรคทอรีที่จะใช้เก็บฟอร์มต้นแบบอื่นอีกอันอาจมีต่อไป เช่นให้เป็น

```
c:\Odelphi\myobjrep
```

4. เก็บไฟล์ .pas (ตามที่ป้อนโปรแกรมไว้) ไว้ในไดเรคทอรีนี้ โดย

- เลือกรายการ File/Save As
- ให้ชื่อไฟล์เป็น Ucgiform.pas

5. เก็บไว้ใน Object Repository โดยเลือกรายการ Project/Add To Repository จะแสดงไดอะล็อกบ็อกซ์ให้ป้อนค่าต่าง ๆ ที่สำคัญดังนี้

- Title ให้ป้อนชื่อที่จะตั้งเป็นชื่อของฟอร์มนี้ ผมป้อนเป็น CGI Script Form ตามที่จะอ้างถึงต่อไปด้วย
- Page ให้เลือกชื่อหน้าเป็น Forms
- ตรงข้อความว่า Select an icon... ให้เลือกปุ่ม Browse จะแสดงไฟล์ไดอะล็อกบ็อกซ์ ให้เลือกไฟล์ .ICO ที่เตรียมไว้ หรือใช้ของเดลไฟเสร็จแล้วให้เลือกปุ่ม OK

6. ยกเลิกโปรเจ็กต์นี้ เช่นโดยเปิดโปรเจ็กต์ใหม่ และปฏิเสธการเก็บลงไฟล์

การนำมาใช้

ในตัวอย่างต่อ ๆ ไปจะนำสกุลของหน้าต่างที่สร้างไว้นี้มาใช้ โดยในคำอธิบายจะอ้างเป็น "ให้นำหน้าต่าง CGI Script Form มาใช้" ซึ่งการใช้ก็คือการสืบสกุลจาก TCgiForm ซึ่งจะดำเนินการได้โดย

1. เปิดโปรเจ็กต์ใหม่

2. เลือกรายการ File/New จะแสดงไดอะล็อกบ็อกซ์ ให้

- เลือกหน้า Forms
- เลือกฟอร์มต้นแบบเป็น CGI Script Form
- เชคปุ่ม Inherit
- เลือกปุ่ม OK

จะทำให้ได้หน้าต่าง CgiForm2 เพิ่มขึ้นอีก 1 หน้าต่าง คือ 1 ฟอร์ม

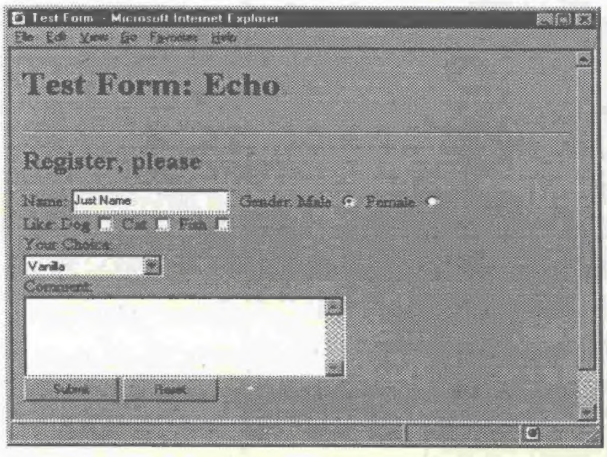
3. ยุบหน้าต่าง Form1 โดยเลือกรายการ View/Project Manager จะแสดงหน้าต่าง Project Manager ให้

- เลือกรายการ Form1
- เลือกปุ่ม Remove Unit คือปุ่มที่มีเครื่องหมายลบ (ให้ปฏิเสธการเก็บลงไฟล์)

หน้าต่าง Form1 จะหายไป เหลือแต่หน้าต่าง CgiForm2 เพื่อการป้อนโปรแกรม ตามที่จะกล่าวในตัวอย่าง

ตัวอย่างที่สอง

จะสร้างเอกสาร HTML เพื่อแสดงแบบฟอร์มตามรูปที่ 4 เมื่อผู้ใช้กรอกแบบฟอร์มแล้วเลือกปุ่ม Submit จะนำข้อมูลไปดำเนินกรรมวิธีตามที่กล่าวมา แต่ตามตัวอย่างนี้จะเพียงแต่ให้ส่งค่ากลับมาเท่านั้น ทั้งนี้เพื่อแสดง



รูปที่ 4 แบบฟอร์มของเอกสาร HTML ตามลิสติงที่ 5

ลิสติงที่ 4

```
unit UCgiForm;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms,
  Dialogs, IniFiles, DB,DBTables;

type
  TCgiForm = class(TForm)
  private
    { Private declarations }
  public
    { Public declarations }
    oIniFi : TIniFile;
    IniFiNme : String;
    OutFi : TextFile;
    function  StartCGIScript : Boolean;
    procedure EndCGIScript;
    procedure WrtTitle(const ttle,bkColor : String);
    procedure WrtHTMLtail;
    procedure WrtText(const txt : String);
    procedure WrtLnText(const txt : String);
    procedure WrtHead(si : Integer; const txt : String);
    procedure WrtHTMLfi(const fiNme : String);
    procedure WrtImage(const imge,attr : String);
    function  GetText(const txtNme : String) : String;
    function  GetTextSect(const sctn,txtNme : String) : String;
    procedure DoTable(oTbl : TTable; headS : array of String;
                                     oFldS : array of TField);
    procedure SaveINIFi(fiNme : String);
    procedure PutDbDtaToHtmlTbl(oDbSet : TDBDataSet);
    procedure PutDbFieldToHtmlCombo(oDbSet : TDBDataSet;
                                    cbNme,fldNme : String);
    procedure StrtForm(exeFiNme : String);
    procedure EndForm(cmmBttNme : String);
  end;

var
  CgiForm: TCgiForm;

implementation

{$R *.DFM}

function TCgiForm.StartCGIScript : Boolean;
var  InFiNme,OutFiNme : String;
begin
 try
  IniFiNme := ParamStr(1);
  oIniFi := TIniFile.Create(IniFiNme);
  Application.ProcessMessages;
 except
  Result := false;
 end;  Result := true;
  OutFiNme := oIniFi.ReadString('System', 'Output File', '');
  AssignFile(OutFi, OutFiNme);  Rewrite(OutFi);
  Writeln(OutFi, 'HTTP/1.0 200 OK');
  Writeln(OutFi, 'Server: WebSite 1.0');
  Writeln(OutFi, 'Content-type: text/html');
  Writeln(OutFi, '');
end;

procedure TCgiForm.EndCGIScript;
begin
  oIniFi.Free;  CloseFile(OutFi);
  Application.Terminate;
end;

procedure TCgiForm.WrtTitle(const ttle,bkColor : String);
begin
```

ลิสติงที่ 4 (ต่อ)

```
    Writeln(OutFi, '<HTML><HEAD>');
    Writeln(OutFi, '<TITLE>' + ttle + '</TITLE></HEAD>');
    Writeln(OutFi, '<BODY BGCOLOR=' + bkColor + '>');
end;

procedure TCgiForm.WrtHTMLtail;
begin
  Writeln(OutFi, '</BODY></HTML>');
end;

procedure TCgiForm.WrtHead(si : Integer; const txt : String);
begin
  Writeln(OutFi, '<H' + IntToStr(si) + '>' + txt +
                 '</H' + IntToStr(si) + '>');
end;

procedure TCgiForm.WrtText(const txt : String);
begin
  WriteLn(OutFi, txt);
end;

procedure TCgiForm.WrtLnText(const txt : String);
begin
  WriteLn(OutFi, txt, '<BR>');
end;

procedure TCgiForm.WrtImage(const imge,attr : String);
begin
  WriteLn(OutFi, '<IMG SRC=" ' + imge + '" ' + attr + '>');
end;

procedure TCgiForm.WrtHTMLfi(const fiNme : String);
var  oTxt : TStringList;  ind : Integer;
begin
  oTxt := TStringList.Create;
  oTxt.LoadFromFile(fiNme);
  for  ind := 0 to oTxt.Count-1  do
    WriteLn(OutFi, oTxt[ind]);
  oTxt.Free;
end;

function  TCgiForm.GetText(const txtNme : String) : String;
begin
  Result := oIniFi.ReadString('Form Literal', txtNme, '');
end;

function  TCgiForm.GetTextSect(const sctn,txtNme : String) : String;
begin
  Result := oIniFi.ReadString(sctn, txtNme, '');
end;

procedure TCgiForm.DoTable(oTbl : TTable; headS : array of String;
                                          oFldS : array of TField);
var  id : Integer;
begin
  WrtText('<TABLE BORDER>');
  WrtText('<TR>');
  for  id := 0 to High(headS)  do
    WrtText('<TD>' + headS[id]);
  oTbl.First;
  while  not oTbl.EOF  do
    begin
    WrtText('<TR>');
    for  id := 0 to High(oFldS)  do
      WrtText('<TD>' + oFldS[id].AsString);
    oTbl.Next;
    end;
  WrtText('</TABLE>');
end;

procedure TCgiForm.PutDbDtaToHtmlTbl(oDbSet : TDBDataSet);
```

- รูปแบบของคอนโทรลทั้งหมดในแท็ก FORM

- การใช้หน้าต่าง CGI Script Form จะเตรียมโปรแกรมได้ดังนี้

เอกสาร HTML

ให้สร้างเอกสาร html ตามลิสติงที่ 5 เก็บไว้ในไฟล์ dmoecho.htm ในไดเรคทอรี myjpage มีรายละเอียดเกี่ยวกับการกำหนดคอนโทรลต่าง ๆ เพิ่มจากตัวอย่างที่แล้ว ซึ่งต้องดูประกอบกับรูปที่ 4 ดังนี้

1. กรอบอีดิต ให้ค่า TYPE เป็น TEXT

2.ปุ่มวิทยุ ให้ค่า TYPE เป็น RADIO และต้องให้ชื่อปุ่มวิทยุที่อยู่ในกลุ่มเดียวกัน (เลือกได้ปุ่มเดียว) เหมือนกัน คือค่าที่ให้แก่ NAME เช่นตามตัวอย่างนี้คือ Gender แต่ให้ค่า VALUE แตกต่างกัน ซึ่งเป็นค่าที่จะได้รับในสคริปต์ เมื่อผู้ใช้เลือกปุ่มใด

3.ปุ่มเช็ค ให้ค่า TYPE เป็น CHECKBOX และต้องให้ค่า NAME กับค่า VALUE แตกต่างกัน เมื่อผู้ใช้เลือกปุ่มใดจะได้ค่า NAME ตามค่า VALUE ถ้าไม่เลือกจะได้ค่า NAME เป็นสตริงว่าง

4. กรอบคอมโบ ใช้แท็กเป็น SELECT และให้ค่ารายการด้วยแท็ก OPTION ซึ่งเมื่อผู้ใช้เลือกรายการใด จะได้ค่า NAME ตามค่ารายการนั้น

5. กรอบเมมโม (memo) ใช้แท็กเป็น TEXTAREA และมีค่าที่จำเป็นต้องให้คือ

- NAME เพื่ออ้างถึง

- COLS และ ROWS เป็นขนาดเป็นจำนวนอักษรใน 1 บรรทัด และจำนวนบรรทัดตามลำดับ

6. ปุ่มควบคุม มี 2 ชนิด คือ

- ทำหน้าที่คล้ายปุ่ม OK ให้ค่า TYPE เป็น SUBMIT

- ที่ทำหน้าที่คล้ายปุ่ม Cancel ให้ค่า TYPE เป็น RESET

```
var  id : Integer;
begin
  if  oDbSet.EOF  then
    begin  WrtText('Search not found');  Exit  end;
  WrtText('<TABLE BORDER>');
  WrtText('<TR>');
  for  id := 0 to oDbSet.FieldCount -1  do
    WrtText('<TH>' + oDbSet.Fields[id].DisplayLabel + '</TH>');
  oDbSet.First;
  while  not oDbSet.EOF  do
    begin
    WrtText('<TR>');
    for  id := 0 to oDbSet.FieldCount -1  do
      WrtText('<TD>' +  oDbSet.Fields[id].AsString);
    oDbSet.Next;
    end;
  WrtText('</TABLE>');
end;

procedure TCgiForm.SaveINIFi(fiNme : String);
var  oTem : TStringList;
begin
  if  fiNMe = ''  then  fiNme := 'c:\mytemp\webini.txt';
  oTem := TStringList.Create;
  oTem.LoadFromFile(IniFiNme);
  oTem.SaveToFile(fiNme);
  oTem.Free;
end;
(* ------- *)

procedure TCgiForm.StrtForm(exeFiNme : String);
begin
  WrtText('<FORM  METHOD="POST" ACTION="/mycgi/' +
                                exeFiNme + '">');
end;

procedure TCgiForm.EndForm(cmmBttNme : String);
begin
  WrtText('<INPUT TYPE="submit" VALUE=' + cmmBttNme + '>');
end;

procedure TCgiForm.PutDbFieldToHtmlCombo(oDbSet : TDBDataSet;
                                    cbNme,fldNme : String);
begin
  WrtText('<SELECT NAME=' + cbNme +'>');
  oDbSet.First;
  while  not oDbSet.EOF  do
    begin
    WrtText('<OPTION> ' + oDbSet.FieldByName(fldNme).AsString);
    oDbSet.Next;
    end;
  WrtText('</SELECT> <BR>');
end;


end.
```

ลิสติงที่ 4 (ต่อ)

CGI Script

ขอให้นำหน้าต่าง CGI Script Form มาใช้ (ตามที่กล่าวมา) โดยป้อนโปรแกรมตามลิสติงที่ 6 สำหรับคำอธิบายโปรแกรมขอให้ดูประกอบกับโปรแกรมในลิสติงที่ 4 ด้วย

และสำหรับคำว่าค่า NAME หมายถึงค่าที่ให้แก่ NAME ในลิสติงที่ 5 ส่วนค่า VALUE คือค่าที่ให้แก่ VALUE หรือผู้ใช้ป้อน หรือเลือกตามเงื่อนไขที่กล่าวมา

จะได้การดำเนินการของโปรแกรมนี้ ซึ่งเป็นคำอธิบายการใช้ CGI Script Form ด้วยดังนี้

ลิสติงที่ 5

```
<HTML>
  <HEAD> <TITLE>Test Form </TITLE> </HEAD>
  <BODY>
<H1> Test Form: Echo</H1> <HR>

<!- ========== Main body ========== ->

<FORM  METHOD="POST" ACTION="/mycgi/dmoecho.exe">
<H2>Register, please</H2>
Name: <INPUT TYPE=text NAME=FirstName VALUE="Just Name">
Gender: Male <INPUT NAME="Gender" TYPE=radio VALUE="Male" CHECKED=True>
             Female <INPUT NAME="Gender" TYPE=radio VALUE="Female"> <BR>
Like:  Dog <INPUT TYPE=checkbox NAME=Like_Dog VALUE=Dog>
          Cat <INPUT TYPE=checkbox NAME=Like_Cat VALUE=Cat>
          Fish <INPUT TYPE=checkbox NAME=Like_Fish VALUE=Fish> <BR>

Your Choice: <BR>
<SELECT NAME=Choice>
<OPTION> Vanilla
<OPTION> Strawberry
<OPTION> Rum and Coke
<OPTION> Peach and Orange
</SELECT> <BR>

Comment: <BR>
<TEXTAREA NAME="UsrComments" COLS=48 ROWS=4> </TEXTAREA> <BR>
<INPUT TYPE=submit VALUE="Submit">
<INPUT TYPE=reset  VALUE=Reset>
</FORM><HR>
 </BODY>
</HTML>
```

1. StartCGIScript และ EndCGIScript เป็นส่วนขึ้นต้นและปิดท้ายของสคริปต์ควรป้อนคู่กัน (เว้นบรรทัดไว้แทรกคำสั่งอื่น)

2. WritTitle และ WrtHTMLTail เป็นส่วนขึ้นต้นและให้ชื่อไตเติลของเอกสาร HTML และส่วนปิดท้ายของเอกสาร HTML ซึ่งควรป้อนคู่กันเช่นกัน

3. WrtHead เพื่อแสดงข้อความที่เป็นหัวข้อ ตามขนาดของฟอนต์ที่เลือกตั้งแต่ 1 ใหญ่สุดถึง 7 เล็กสุด

4. WrtText และ WrtLnText เพื่อแสดงข้อความธรรมดา โดยที่ WrtLnText ให้ขึ้นบรรทัดใหม่ และสำหรับสตริงที่ให้แสดงอาจประกอบด้วยแท็ก ของ HTML ได้

5. GetText เป็นฟังก์ชันให้ค่า VALUE ตามค่า NAME

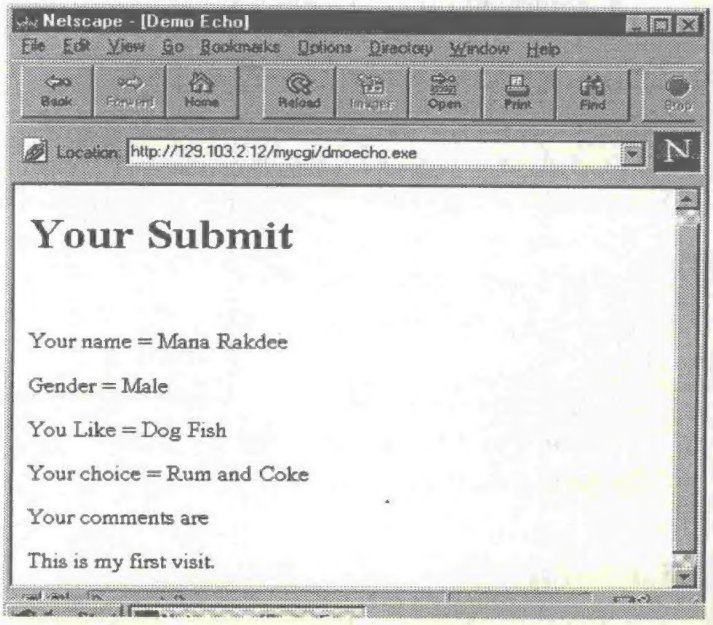
ผลของโปรแกรมนี้จะได้เป็นเอกสาร HTML ตามรูปที่ 5

กิจกรรมในสกุล TCgiForm ที่ไม่ได้นำมาใช้ในโปรแกรมนี้ได้แก่

6. WrtImage เพื่อการแสดงรูปภาพ

7. WrtHTMLfi ใช้ในกรณีที่ต้องการแสดงข้อความในไฟล์ .html ทั้งไฟล์ ให้ขึ้นต้นและปิดท้ายด้วย StartCGIScript และ EndCGIScript

8. PutDbDtaToHtmlTbe จะนำข้อมูลใน TTable หรือ TQury ที่ผ่านมาให้แก่ oDbSet ออกแสดงในรูปแบบของ TABLE คือตารางของ HTML ซึ่งจะต้องจัดจำนวนเรคอร์ด และฟิลด์ให้เหมาะสมด้วย



รูปที่ 5 แบบฟอร์มของเอกสาร HTML ตามที่สร้างด้วยโปรแกรมในลิสติงที่ 6

ลิสติงที่ 6

```
unit DmoEcho1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  UCGIFORM;

type
  TCgiForm2 = class(TCgiForm)
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  CgiForm2: TCgiForm2;

implementation

{$R *.DFM}

procedure TCgiForm2.FormCreate(Sender: TObject);
begin
  inherited;
  StartCGIScript;
  WrtTitle('Demo Echo', 'F0F0F0');
  WrtHead(1, 'Your Submit');

  WrtLnText('Your name = ' + GetText('firstName'));
  WrtLnText('Gender = ' + GetText('Gender'));
  WrtLnText('You Like = ' + GetText('Like_Dog') + ' '
                          + GetText('Like_Cat') + ' '
                          + GetText('Like_Fish'));
  WrtLnText('Your choice = ' + GetText('Choice'));
  WrtLnText('Your comments are');
  WrtText(GetText('UsrComments'));

  WrtHTMLTail;
  EndCGIScript;
end;

end.
```

9. SaveINIFi เพื่อเก็บไฟล์ INT ไว้ศึกษา

10. กิจกรรมที่เหลือเป็นการสร้างฟอร์มให้ผู้ใช้กรอก ซึ่งจะขอนำไปอธิบายเมื่อนำมาใช้

ท้ายตอน

ในฉบับนี้คงพอทำให้คุณได้สัมผัสกับ Intranet ได้ตามสมควร ตอนต่อไปครับจะเป็นเรื่องของการโปรแกรมกับฐานข้อมูล

แต่ผมมีเรื่องส่วนตัวที่จำเป็นต้องเรียนให้ทราบว่า ด้วยสุขภาพของผมขณะนี้ บทความของผมอาจขาดหายไปบ้าง ขออภัยและขอบคุณครับ

ข่าวดีสำหรับผู้ที่สนใจเดลไฟล์

ผู้ที่สนใจเดลไฟล์ แต่ยังไม่มี บอร์แลนด์ให้ดาวน์โหลดมาทดลองใช้ได้ที่

www.borland.com